

4.1.5.1.2 Periodic checking and propagation of global information Another potential important role of the IAM is to monitor the new information in remote JiNao MIBs for a global intrusion detection system. Once such information is available, IAM will retrieve them from the remote global JiNaoMIB, convert them into the format expected by the LDecM, and pass them to the LDecM.

4.1.5.2 Interface Mechanisms and Formats Information exchange on the interface between IAM and LDecM is expected to use message queues in the implementation. In particular, each module will have an input queue for every input interface, onto which other modules supplying inputs will deposit messages. The receiving module will remove messages from these queues and act upon them accordingly. For the interface formats, please see Section 2.3.

4.1.5.3 Scope of Impact Representation In Section 2.3, there is an entry under IAM2MIB which conveys the scope of impact from the decision module. More discussion is in order. Scope of impact information is used by a set of distributed JiNao decision modules in order to better enhance the accuracy of intrusion detection decisions. For example:

- a local decision module (LDecM) may use global information on a power outage to reduce its own false alarm rate with respect to a neighbor router if it knows that the neighbor falls within the scope of impact of the outage;
- a higher-level decision module, i.e. one that has access to observations on a larger topological region, can correlate multiple detections from lower-levels according to their respective scope of impact, and to reach a more accurate detection decision.

To support the objective of a scalable intrusion detection capability, it is important to include the topological information on all the routers affected as part of the scope of impact data. One reasonable representation will be a graph that identifies the routers as nodes, inter-router links as edges, along with the unique identifiers for the routers. In addition, we can also include information such as OSPF adjacency in the data structure. Such information seems useful for making intrusion detection decisions in general. If we allow the possibility of topological information being compiled from a different time than the detection decision, we also need a timestamp associated with the topology graph. Otherwise, the timestamp associated with the detection decision should be sufficient.

4.1.6 Management Information Base (MIB)

JiNao Management information base (JiNaoMIB) is a standard abstraction interface between the JiNao agent and the management applications that are interested in utilizing the intrusion detection services provided by JiNao. The management applications, which will be discussed in the next section, will interact with this MIB through an SNMP engine. This engine will receive and process SNMP PDUs and forward them appropriately to different MIBs. It is also possible for the SNMP engine to communicate the MIB module through so-called *agent extension* protocols. In other words, a MIB access request, as shown in Figure 1, will first pass through a SNMP channel and then pass through the agent extension channel

before it can get to the target MIB. Currently, at least three agent extension protocols have been defined: SMUX[8], DPIv2[9], and more recently AgentX[10].

JiNaoMIB includes five different sections:

Rule/FSM Configuration: Rules and FSMs are used in prevention module as well as local detection module, and they are dynamically loadable from a trusted remote management application. Therefore, the JiNaoMIB specification provide an interface to support this feature.

Each rule in the prevention is represented as a MIB table entry with a set of attributes. By accessing these attributes, we can activate and deactivate the rule. It is also possible to adjust the thresholds used by the rules. All these rule table-entries are placed in a table called "JiNaoPrevRuleTable." Similarly, we will have a table called "JiNaoDetectFSMTable" for all the FSMs used by the local detection module.

Please note that SNMP does not support "direct" table entry insertion and deletion. We will use another MIB variable to indirectly achieve these two unsupported operations. Furthermore, we are expecting to use a secure version of SNMP (like SNMPv2* or SNMPv3/ng). Thus, we need to worry about not only authentication and integrity but also access control. Most of the MIB variables should be restricted to trusted security managers. We should not even allow normal users having read-only access to the rule/FSM tables. If we allow this to happen, then potential intruder will know what attacks JiNao is trying to prevent/protect against, so they can avoid being detected. Therefore, all access to the JiNao MIB must be authenticated and encrypted.

Local Detection Results: The local decision module, after performing analysis on the events or messages, makes decisions about whether certain intrusion attacks have happened. This information is very valuable and should be accessed by trusted security management applications through the MIB interface.

Each piece of information should be represented as a JiNao report table entry (i.e., JiNaoReportTableEntry). All the reports together form the JiNaoReportTable. This table is updated by JiNao decision module periodically to reflect the health of the neighbor routers in real-time. This table is "read-only" through the SNMP/DPIv2 interface. Again, access to the information in JiNaoReportTable must use an authenticated and encrypted message channel with appropriate access control.

Detection Notifications: A particular trusted security management application might be interested in knowing if one particular type of report has been updated in the MIB. Traps/Event notifications are very useful in this situation. In JiNao, this security management application can express its interest in certain types of information through the SNMP MIB interface. The JiNao agent, upon receiving the request, will start to generate traps/events for the application when an event occurs.

Security Control: Most of the security control actions can be achieved by inserting or deleting the rules or FSMs through the rule/FSM configuration MIB section. However, there are cases where we must provide other control interfaces to achieve the goal. For example, a security control console will be directly connected to the local decision

module. A trusted system administrator will use the console to access the JiNao information and control the system directly. Sometimes, a remote management module might want to directly notify the administrator through the MIB interface.

Log Access: In the prevention module, selected PDUs are logged in an audit trail. These logged PDUs may be valuable for some off-line analysis, and they should be accessed through the SNMP protocol.

Apparently, it is unrealistic to provide a unique object identity for each individual record. A search/query engine directly working on the log database is necessary. We merely use the SNMP MIB interface to control this search/query engine. For example, if a management application is interested in receiving all the OSPF PDUs originated by a particular router, then it will, through the SNMP SET/GET interface, submit a query to the search engine. The search engine will retrieve all the matched entries and put them in a table called JiNaoLogAccessResultTable. Then, the requesting security management application can use SNMP Get/GetNext/GetBulk to retrieve all the records in that table.

4.2 Management Information Exchange Protocol

For interoperability, we chose SNMP as the management information protocol to exchange control/management information among the distributed entities in the JiNao system. The current standardized version of SNMP is still version 1 which is not secure (or security was not a concern in version 1). There are at least two proposals for SNMPv2 security: SNMPv2* and SNMPv2u. The new IETF working group: SNMPng (Simple Network Management Protocol: Next Generation) was just established in March 1997. The mission of this working group is to unify different security proposals and to come out with one simple and secure SNMP framework.

The current framework supports two levels of security: message-level security and local process access control. The former concerns a secure (Authentication, Integrity, and Privacy) channel between the MIB agent and an authenticated user. The latter is for capability and resource access control. For example, a normal user's SNMP request for removing a route entry will pass the message-level security, but will be denied by the access control mechanism in local processing module (LPM).

The current security framework of SNMPv3/ng only covers the SNMP PDUs themselves. It does not cover the security concerns for subagent protocols like SMUX, DPlv2, and AgentX. The rationale for this is that the security checks would have been performed by the master agent in the SNMP level. This rationale is fine if the master and subagents are on the same node running a secure OS or both located in a private network segment. However, if it is connected through a public network, the security is an important consideration. For example, we can use any secure transport layer protocol to secure the channel between the master and subagent.

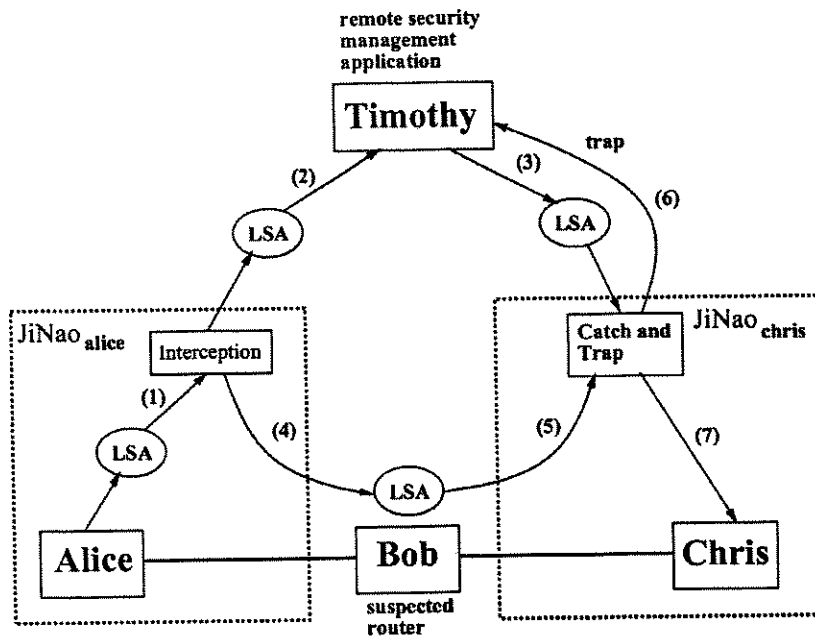


Figure 4: Figurative description of catch and trap.

4.3 Remote JiNao Management Applications

Through the SNMPv3/ng interface, the JiNao IDS service is available to all authenticated and authorized SNMP-based application entities over the public Internet. A remote management station can access the JiNao MIB on different routers and correlate these distributed JiNao results. A MIB specification will be defined such that the remote management applications can interpret the MIB information correctly. In this section, we give an example of remote intrusion detection with a Catch-and-Trap MIB interface.

A remote JiNao management application would like to find out if one particular router is not processing the input protocol data units faithfully. For example, a compromised OSPF router modifies the LSA (Link-State information) originated by this router. If a digital signature scheme is NOT used, it is hard to detect such an attack by one single JiNao. One way to detect this intrusion is to compare the input LSA with output LSA from this compromised router. This can be done by two different approaches using the JiNao MIB interface we just described:

JiNao Log Access MIB: By delegating the proper rules into the prevention module, various types of OSPF PDUs can be logged and retrieved from the Log Access MIB interface. Please note that the prevention module can not only log the incoming PDUs but also the outgoing ones. A remote management application can access the logs on two neighbors of this suspected router. Then, by comparing the log files, the remote management model can tell whether the LSAs have been faithfully forwarded.

JiNao Catch and Trap MIB: Checking and comparing log files might take certain amount

of time, communication and computation resources because the log could contain a large amount of information. Ideally, if the comparison task is performed in the prevention module itself, it will be much more efficient.

The following example is used to describe the functionality of the catch and trap interface: *Alice*, *Bob*, and *Chris* are routers connected to one another as shown in Figure 4. The remote management application *Timothy* is suspecting that *Bob* has been compromised. *Timothy* will send a *suspend* request to the out-going prevention module of *Alice* to catch and hold one outgoing LSA (LSA_x , which should be sent to *Bob*). Now, *Timothy* will use the Catch and Trap interface on *Chris*. After the request, *Chris* knows that he should look at all the OSPF PDUs from *Bob* and check if one of them is LSA_x . At this point, *Timothy* will notify *Alice* to release LSA_x . Now, if *Chris* catches LSA_x , he will trap/notify *Timothy* immediately. If, after δ amount of time, he can not find LSA_x , he will also notify *Timothy* with a Catch-failure report. This catch-and-trap MIB interface facility can be used to efficiently handle compromised routers.

References

- [1] J. Anderson, "Computer Security Threat Monitoring and Surveillance", Fort Washington, PA: James P. Anderson Co., April 1980.
- [2] J. Winkler, "A UNIX Prototype for Intrusion and Anomaly Detection in Secure Networks", Proceedings, 13th National Computer Security Conference, Oct. 1990.
- [3] D. Anderson, T. Frivold, and A. Valdes, "Next-generation Intrusion Detection Expert System (NIDES), A Summary", Technical Report SRI-CSL-95-07, Computer Science Laboratory, May 1995.
- [4] D. E. Denning, "An Intrusion-Detection Model", IEEE Transactions on Software Engineering, Vol. 13, No. 2, Feb. 1987.
- [5] R. Jagannathan and T. Lunt, "System Design Document: Next Generation Intrusion Detection Expert System (NIDES)", SRI report, SRI International, Menlo Park, CA, March 9, 1993.
- [6] L. T. Heberlein, *et al.*, "Internetwork Security Monitor: An Intrusion-Detection System for Large-Scale Networks", Proceedings, 15th National Computer Security Conference, Oct. 1992.
- [7] K. Jackson, D. DuBois, and C. Stallings, "An Expert System Application for Network Intrusion Detection", Proceedings, 14th National Computer Security Conference, Oct. 1991.
- [8] M. Rose, "RFC1227: SNMP MUX Protocol and MIB", May 23, 1991.
- [9] B. Wijnen, G. Carpenter, K. Curran, A. Sehgal, G. Waters, "RFC 1592: Simple Network Management Protocol Distributed Protocol Interface Version 2.0", March 3, 1994.

- [10] M. Daniele, B. Wijnen, and D. Francisco, "Agent Extensibility (AgentX) Protocol, Version 1", Internet Draft, Nov 26, 1996.
- [11] H. S. Javitz and A. Valdes, "The NIDES Statistical Component Description and Justification", Annual Report A010, SRI, March 7, 1994.
- [12] "Wisdom and Sense Guidebook", Los Alamos National Laboratory, Los Alamos, New Mexico.
- [13] S. E. Smaha, "Haystack: An Intrusion Detection System", Proceeding IEEE Fourth Aerospace Computer Security Applications conference, Orlando, FL, Dec. 1988.
- [14] R. Cleaveland, J. Parrow, and B. Steffen, "The Concurrency Workbench: A Semantics-Based Tool for the Verification of Finite-State Systems", ACM Transactions on Programming Languages and Systems, Vol 15, No. 1, pp 36-72, Jan. 1993.

Internet Options

Security Settings

Settings:

NET Framework-reliant components

- ☒ Run components not signed with Authenticode
 - ☐ Disable
 - ☐ Enable
 - ☐ Prompt
- ☒ Run components signed with Authenticode
 - ☐ Disable
 - ☐ Enable
 - ☐ Prompt
- ☒ ActiveX controls and plug-ins
 - ☒ Download signed ActiveX controls
 - ☐ Disable
 - ☐ Enable
 - ☐ Prompt

Reset custom settings

Reset to: **Medium-low**

Total UB 108,081.36 **Amount Billed (Std Value)** 56,307.00

Realization Ratio 29.27 **Realization Ratio (Std)** 28.64

Fees Received 59,347.42 **Costs Written Down/Up** 0.00

Done

Start Elite WebView - Micro...

Internet Options

Security Settings

Settings:

☒ Download unsigned ActiveX controls

- ☐ Disable
- ☐ Enable
- ☐ Prompt

☒ Initialize and script ActiveX controls not marked as safe

- ☐ Disable
- ☐ Enable
- ☐ Prompt

☒ Run ActiveX controls and plug-ins

- ☐ Administrator approved
- ☐ Disable
- ☐ Enable
- ☐ Prompt

Reset custom settings

Reset to: **Medium-low**

Total UB 108,081.36 **Amount Billed (Std Value)** 56,307.00

Realization Ratio 29.27 **Realization Ratio (Std)** 28.64

Fees Received 59,347.42 **Costs Written Down/Up** 0.00

Done

Start Elite WebView - Micro...

Elite WebView - Microsoft Internet Explorer

Internet Options

Security Settings

Settings:

- ☒ Script ActiveX controls marked safe for scripting
 - ☐ Disable
 - ☐ Enable
 - ☐ Prompt
- ☒ Downloads
 - ☒ File download
 - ☐ Disable
 - ☐ Enable
 - ☒ Font download
 - ☐ Disable
 - ☐ Enable
 - ☐ Prompt
- ☒ Miscellaneous
 - ☒ Access data sources across domains

Reset custom settings

Reset to:

Done

Start | Elite WebView - Micro...

Number 00307 Rate 695.00

06 New Period 05/06 01/06-05/06

Up	146.20	717.50
	(0.60)	(6.30)
	100.135.36	491.475.86
	0.00	20.00
(Value)	55.896.00	451.393.87
(al Value)	56.307.00	457.623.50
	56.307.00	457.503.50
	29.27	28.64
	29.27	28.66
	59.347.42	390.652.35
	0.00	(1.065.97)

Total UB 108,081.36 Amount Billed (Std Value)

Realization Ratio 29.27 28.64

Realization Ratio (Std) 29.27 28.66

Fees Received 59,347.42 390,652.35

Costs Written Down/Up 0.00 (1,065.97)

Local intranet

Elite WebView - Microsoft Internet Explorer

Internet Options

Security Settings

Settings:

- ☒ Miscellaneous
 - ☒ Access data sources across domains
 - ☐ Disable
 - ☐ Enable
 - ☐ Prompt
 - ☒ Allow META REFRESH
 - ☐ Disable
 - ☐ Enable
 - ☒ Allow scripting of Internet Explorer Webbrowser controls
 - ☐ Disable
 - ☐ Enable
 - ☒ Display mixed content
 - ☐ Disable
 - ☐ Enable

Reset custom settings

Reset to:

Done

Start | Elite WebView - Micro...

Number 00307 Rate 695.00

06 New Period 05/06 01/06-05/06

Up	146.20	717.50
	(0.60)	(6.30)
	100.135.36	491.475.86
	0.00	20.00
(Value)	55.896.00	451.393.87
(al Value)	56.307.00	457.623.50
	56.307.00	457.503.50
	29.27	28.64
	29.27	28.66
	59.347.42	390.652.35
	0.00	(1.065.97)

Total UB 108,081.36 Amount Billed (Std Value)

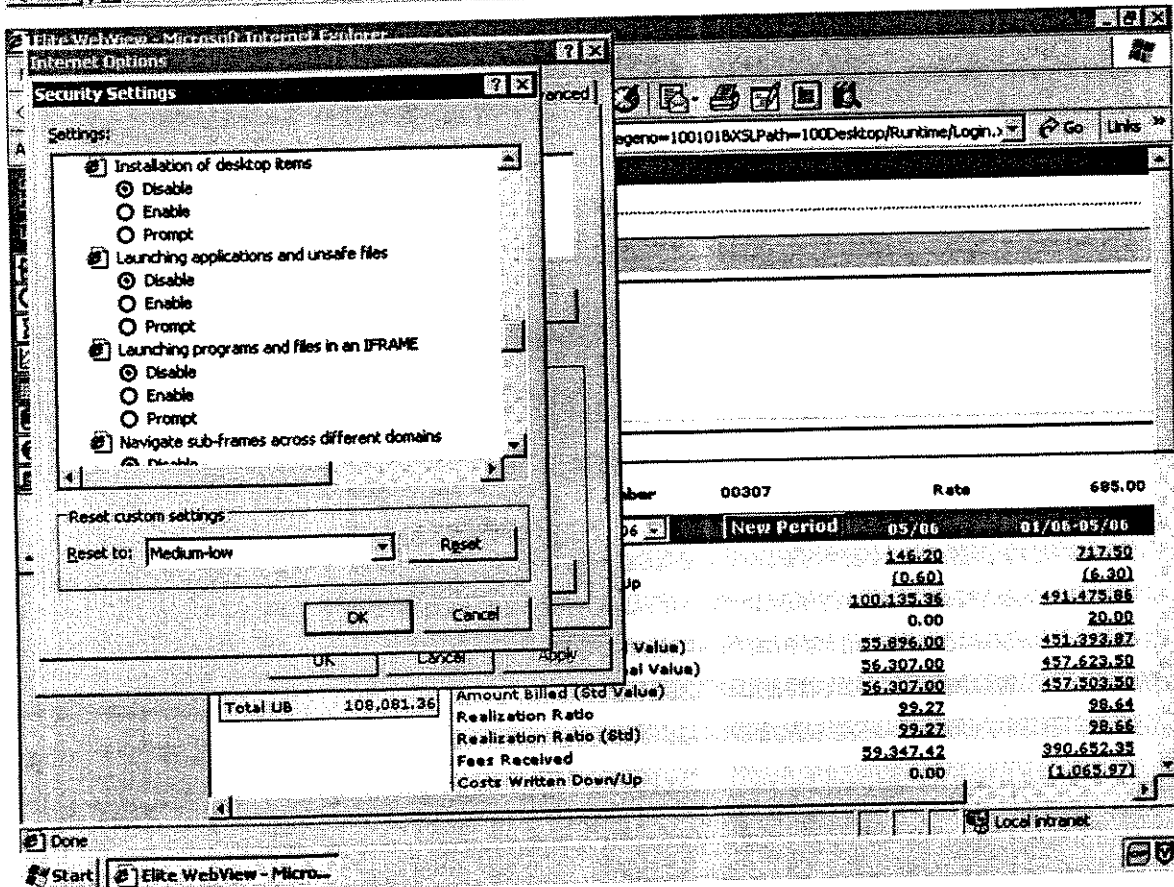
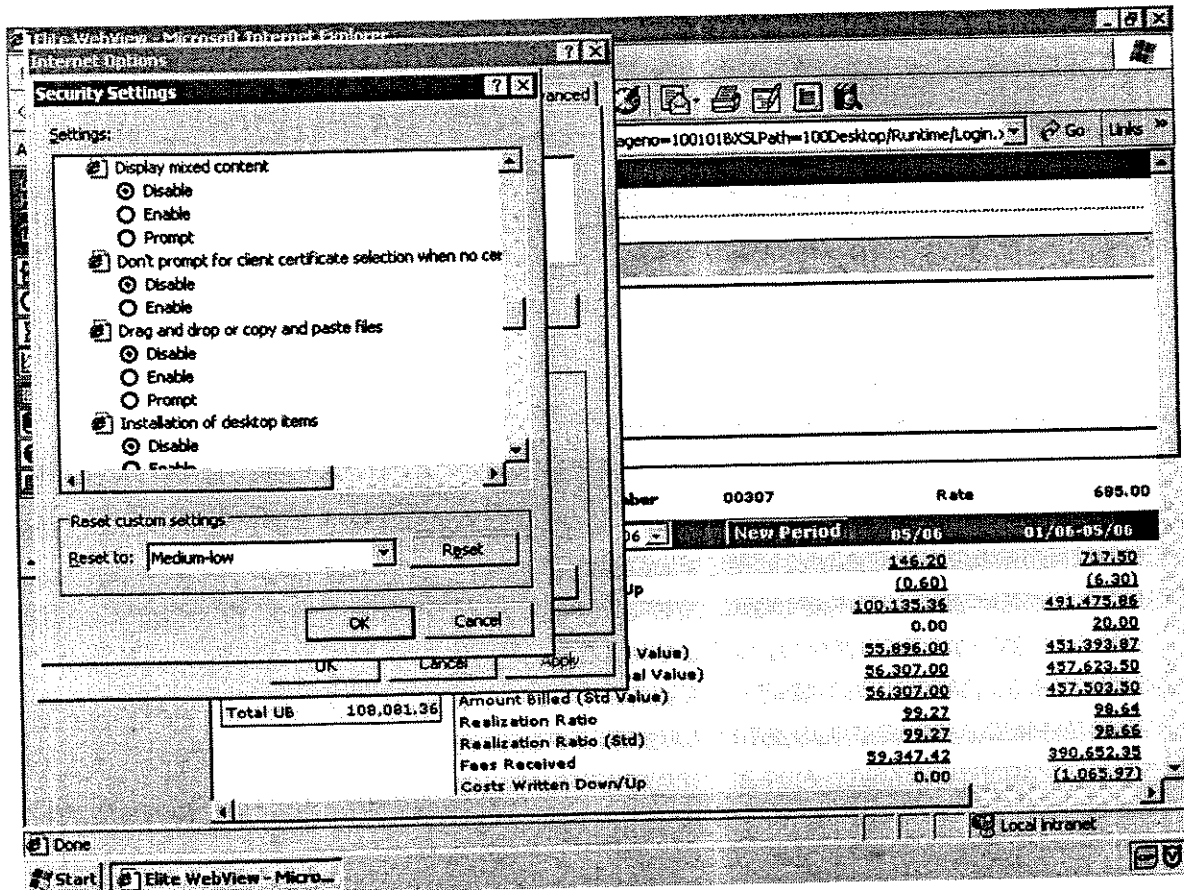
Realization Ratio 29.27 28.64

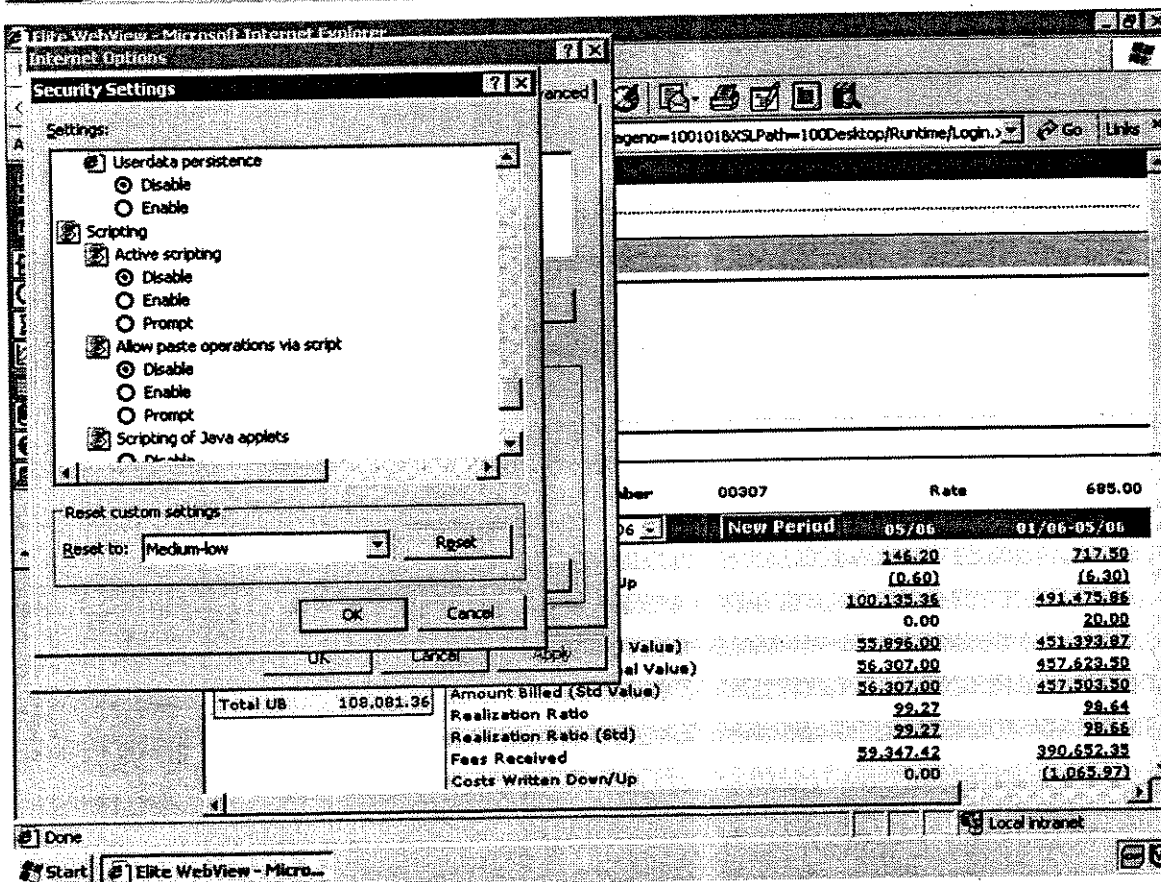
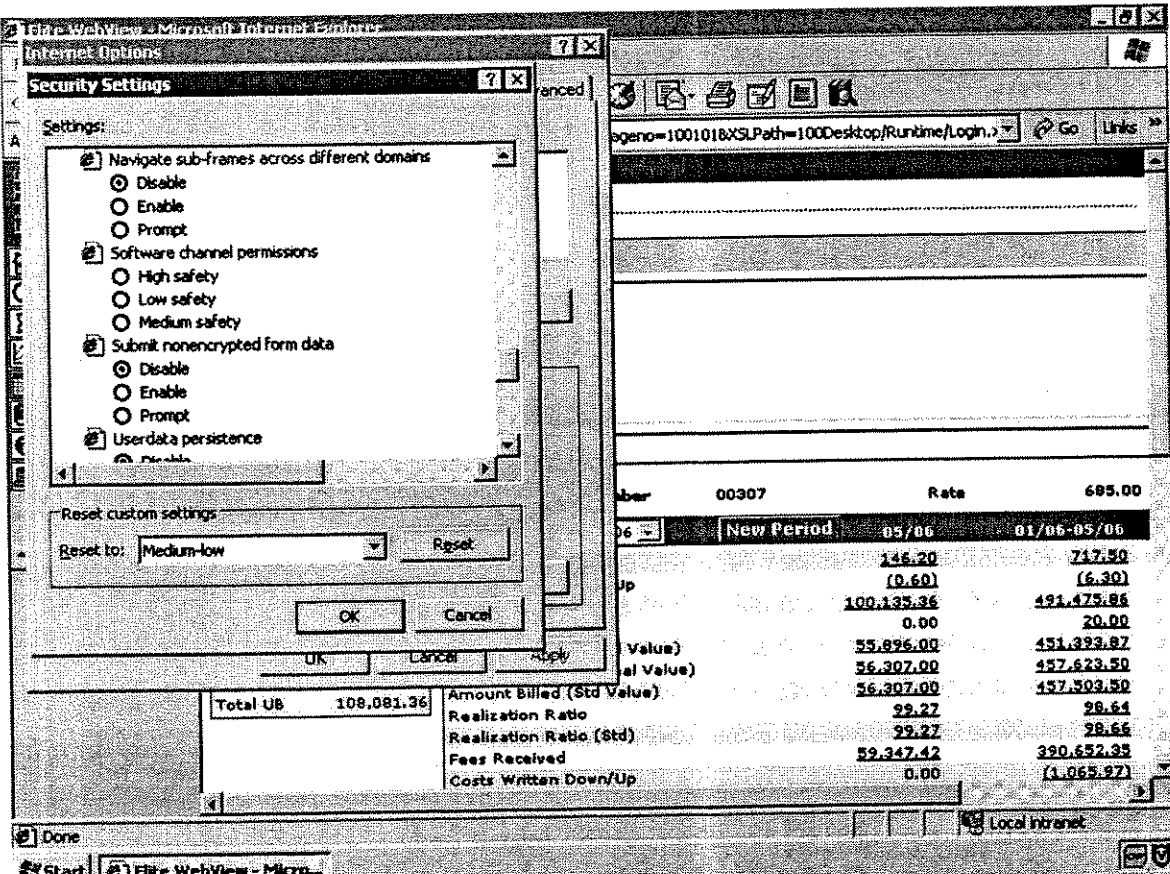
Realization Ratio (Std) 29.27 28.66

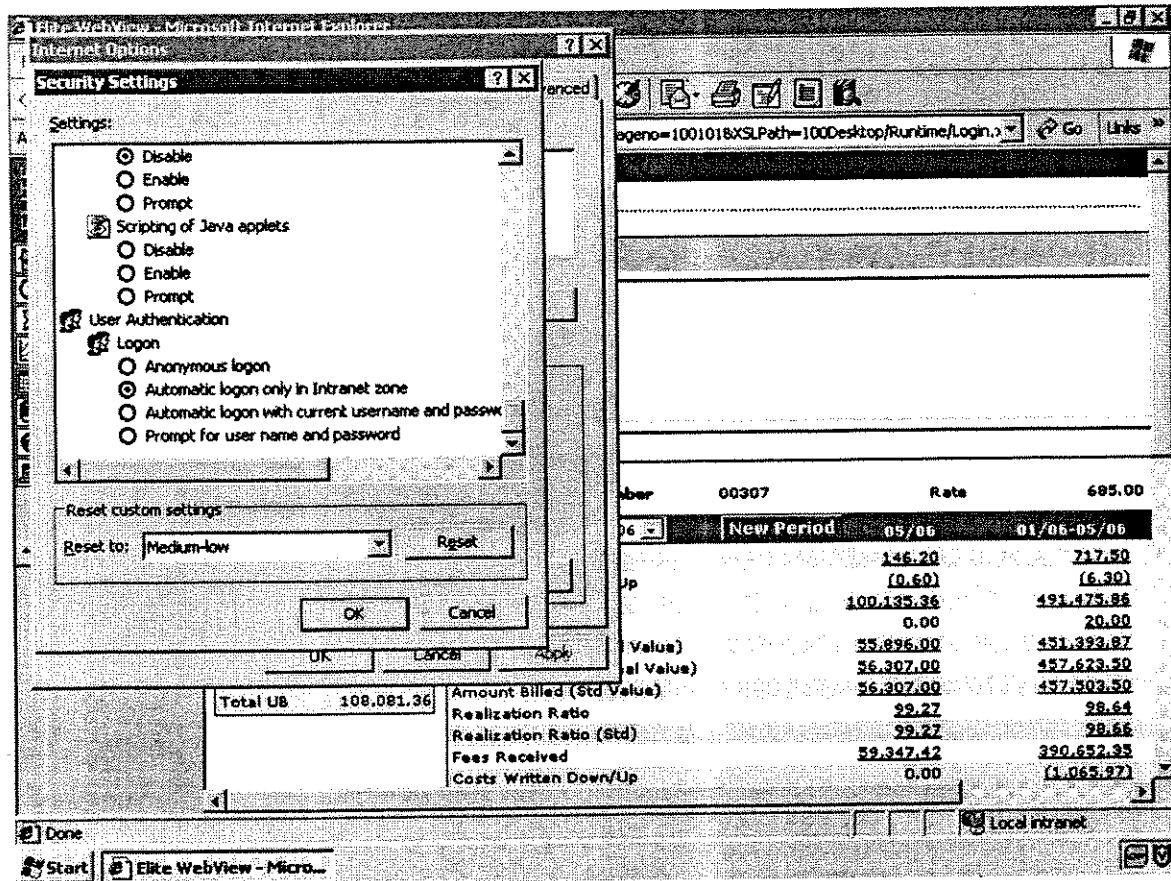
Fees Received 59,347.42 390,652.35

Costs Written Down/Up 0.00 (1,065.97)

Local intranet







IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

C.A:04-1199 (SLR)

SRI INTERNATIONAL, INC.,)
a California Corporation)
)
Plaintiff and)
Counterclaim Defendant,)
)
v.)
)
INTERNET SECURITY SYSTEMS, INC.,)
a Delaware Corporation, INTERNET)
SECURITY SYSTEMS, INC., a Georgia)
Corporation, and SYMANTEC)
CORPORATION, a Delaware)
Corporation,)
)
Defendants and)
Counterclaim-Plaintiffs.)
-----)

COPY

VIDEOTAPED DEPOSITION

OF

Y. FRANK JOU

At Raleigh, North Carolina
January 27, 2006 - 9:53 a.m.

Reported by:
Debra D. Bowden

capitalreporting

PO Box 97696
Raleigh, NC 27624

8360 Six Forks Road
Suite 101
Raleigh, NC 27615

919.398.7775 ph
919.398.7741 fax

www.capreporting.com

capreporting@aol.com

1 meant -- what I meant was the capability we
2 implement was local in nature.

3 Q. Um-hmm.

4 A. And as the goal we try to achieve was to be
5 able to scale this capability to a global
6 label. So that was my intent in this
7 description here. Basically as a next step
8 in the capability it should be extend from
9 local to a global area. Global scope.

10 Yeah.

11 Q. Okay. And now the DARPA project was a
12 three-year project; correct?

13 A. Right.

14 Q. It was a limited in time; correct?

15 A. Yeah, um-hmm.

16 Q. And limited in funding money; correct?

17 A. Yeah.

18 Q. Had you had more time and money, would you
19 have taken that natural extension step to a
20 more global system?

21 A. Definitely that was in our intent. But you
22 know, again I should say this was a
23 research project. There was no guarantee,
24 you know, we would be able to bear any

1 fruit even though if the time or resource
2 is allowed at that point in time.

3 Q. If you go back to the architecture
4 document, J18, on page 3.

5 A. Page 3. Okay.

6 Q. And if you go to Section 2.1.

7 A. Um-hmm.

8 Q. And you go to the third paragraph.

9 A. Um-hmm.

10 Q. The middle of it. And you say, "While it
11 is not within the scope of this project, we
12 expect that the detection analysis
13 functions implemented in the local
14 subsystem can be extended to a global level
15 and correlate intrusion events among
16 several routers." Do you see that?

17 A. Um-hmm.

18 Q. And then it goes on to say, "The management
19 capability which is based on SNMP framework
20 can logically be further extended among
21 management nodes in a hierarchical fashion
22 to establish a status map for an autonomous
23 system."

24 A. Um-hmm.

1 Q. Now, while your DARPA project was limited
2 in time and funding, did you create the
3 design such that it could be extended in
4 this hierarchical fashion?

5 A. I would not say created, because the SNMP
6 network by its nature is to monitor remote
7 system.

8 Q. Um-hmm.

9 A. And be able to reflect a healthy -- the
10 healthy -- the status of the network, you
11 know, it's healthy, whether it's healthy or
12 it's, you know, under stress. That was the
13 intent of the SNMP framework. And our
14 thinking at that point in time was to take
15 advantage of this SNMP by the fact that
16 it's able to monitor several systems in a
17 distributive fashion. And you know, the
18 challenge at that point was how do you
19 correlate. I think that was the main
20 technical challenge at that point in time
21 in terms of how do you collect -- collect
22 of the local detection result was not an
23 issue. The issue was how do you come up
24 with the intelligence, how do you correlate

1 all the relevant information and be able
2 to, you know, derive a certain logical or
3 reasonable conclusion, and able to, based
4 upon this result, take action accordingly.
5 I think that was the challenge, and the --
6 you know, we did look into that aspect.
7 But however at that point we did not have a
8 very promising, you know, development at
9 that time. At the conclusion of the
10 project. So that was, you know, the open
11 question at that point.

12 Q. And if you just saw the term correlate --

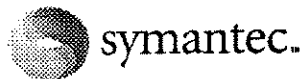
13 A. Um-hmm.

14 Q. -- what would that mean to you?

15 MS. PRESCOTT: Objection to form.

16 A. Correlate means how do you put two or more
17 than two input together and derive
18 meaningful information, or intelligence,
19 out of these different infrastreams of
20 information, and be able to come up with
21 certain rationale or logic that what this,
22 you know, behavior manifests to itself.

23 Probably that's kind of lengthy or
24 wordy, but that's my understanding of this



Introduction

There is no silver bullet

The fact is, no intrusion detection method is a panacea, each has strengths and weaknesses. Signature-based approaches can miss new attacks; Protocol Anomaly Detection can miss attacks that are not considered anomalies; Traffic Anomaly Detection misses single-shot or low volume attacks; and, finally, Behavioral Anomaly Detection misses attacks that are difficult to differentiate from normal behavior.

The key to choosing the right IDS is to look for one that combines multiple techniques and technologies into a complete solution. In addition, the product should adapt to the changing threat landscape by adopting new techniques and technologies that either improve upon or replace existing ones.

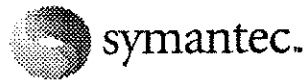
Symantec ManHunt currently uses a variety of trusted and tested methods of detecting network-borne threats. These include Protocol Anomaly Detection (PAD), Traffic rate monitoring, and network pattern matching (signatures).

In addition, users can increase the detection capabilities by using Flow Alert Rules and custom signatures. Flow alert rules allow users to monitor network policy by using the Symantec ManHunt response capabilities to respond to traffic to or from IP address and port combinations. Custom signatures allow users to add network patterns to the supported set that are specific to a particular network environment. Examples include monitoring proprietary protocols, searching for honey-tokens or detecting disallowed application versions.

Protocol Anomaly Detection Techniques

Symantec ManHunt's Protocol Anomaly Detection (PAD) is a form of anomaly detection. Anomaly detection detects threats by noting deviations from expected activity, rather than known forms of misuse. Simply put, anomaly detection looks for "good traffic" and alerts when it does not see it. This is the compliment of a signature-based approach which looks for what is known to be "bad traffic".

Symantec ManHunt provides in-depth models of the most frequently used network protocols. This provides extensive detection capability that goes well beyond simpler forms of protocol analysis. These models provide much deeper detection and fewer false positives because they are able to follow a client server exchange throughout the life of the connection. For example, if a protocol defines the size of a field and ManHunt sees a field that is longer an alert is triggered. Further, if normal usage of the protocol is for much smaller data exchanges and ManHunt sees a very large exchange that would be



considered allowed by the documented specification it will still trigger an alert because the traffic violated expected behavior.

Symantec ManHunt 3.0 has overcome one of the major issues surrounding PAD, overly generic alerts. During a zero day attack a general PAD alert is often all that is possible. However, soon after a new threat is discovered it is often identified by a name and assigned a unique identifier by authorities such as SecurityFocus and CERT. These organizations publish descriptions of the threat and provide pointers to vendor patches or other remediation tools. When this happens it is better to have specific threat identification instead of a protocol anomaly alert. ManHunt now provides event refinement to address this issue. Threats identified by PAD are further analyzed to determine if they are known or unknown. This processing is done after the traffic has been identified and recorded so it need not interfere with the detection performance. This provides the high performance of PAD with the granular identification of a signature matching engine.

Traffic Rate Monitoring

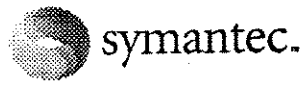
Symantec ManHunt performs passive traffic monitoring on its detection interfaces. It then uses this data to perform both aggregate traffic analysis and individual packet inspection. Individual packets are inspected and traffic is analyzed on a per interface basis. In addition, it uses Netflow data that is either locally collected or forwarded from a remote device to augment its traffic analysis.

Symantec ManHunt's aggregate analysis is able to detect both denial of service and distributed denial of service attacks. These attacks are recognized as unusual spikes in traffic volume. Using the same data, ManHunt can also recommend proper remediation of the problem.

Beyond just attack detection, ManHunt uses traffic analysis to detect many information gathering probes. Heuristics are used to detect not only common methods of probing but also many stealth modes that slip through firewalls and other defenses. For example, many firewalls reject attempts to send SYN packets but allow FIN packets. This results in a common method of doing port scans. ManHunt recognizes the anomaly and triggers an alert.

Network Pattern Matching

ManHunt uses network pattern matching or signatures to provide an additional layer of detection. Signature detection is defined as detecting threats by looking for a specific pattern/fingerprint of a known 'bad' thing. This known-bad pattern is called a signature. These patterns are traditionally based on the observed network behavior of a specific tool or tools.



Signature detection operates on one basic premise. Each threat has some observable property which can be used to uniquely identify it. This property may be based on any property of the particular network packet or packets that carry the threat. In some cases, this may be a literal string of characters found in one packet or it may be a known sequence of packets that must be seen together. In any case, every packet is compared against the pattern. Matches trigger an alert, while failure to match is processed as non-threatening traffic.

Symantec ManHunt uses signatures as a compliment to PAD. The combination provides robust detection without the weaknesses of either PAD alone or signatures alone. ManHunt's high performance is maintained by matching against the smallest set of signatures as is possible given the current context. Since many threats are detected and refined through the PAD engine ManHunt minimizes the set of required signatures to maximize performance.

Future Directions

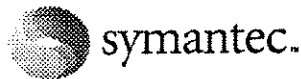
The threat landscape is always changing. In response to this, Symantec is always researching new techniques and improvements to existing ones. A comprehensive list is impossible to foresee. Here is a sneak peek at techniques that may be available in the coming 18 months.

Port Usage Anomaly

Symantec ManHunt will include a port usage anomaly engine which tracks the established connections and connection attempts on a network and looks for patterns indicative of worms, backdoors, and many information gathering techniques.

Many threats reveal themselves by using uncommon ports for illicit purposes. These can be backdoors into system control functions or command and control channels of other kinds. For example, the classic remote control tool, BackOrifice, listens for connections to the victim host on port 31337. This port is not usually used on a production network. The W32.Beagle.E mass mailing trojan horse opens a backdoor on port 2745. Both of these cases are examples of threats that will be detected by future versions of ManHunt before they are recognized or named by security authorities. That is 0 day detection.

In addition, beyond unusual ports many threats create anomalous patterns of connections to common ports. Most worms follow a common pattern of behavior. They scan a network for vulnerable machines, compromise them, install the worm process and begin scanning from the new machine. This behavior will be recognized as such because the scan will attempt connections to hosts not running the vulnerable service. The infamous SQL Slammer worm exhibited this anomalous behavior as it broadcast itself to all hosts



on a network independent of whether the targets were vulnerable. Likewise, the Blaster worm propagated via the Microsoft RPC port 135 in a scan->exploit->scan pattern.

Evasion Profiling

Symantec continues to research attacker tools, techniques, and direction. One such area is that of IDS evasion. Symantec ManHunt is already resistant to many forms of evasion. Future version will include revolutionary new techniques to not only detect an attempt to evade but also the intended goal of the evasion.

This will be accomplished at multiple points. The first is at the network sensor, where knowledge of the network topology will allow Symantec ManHunt to determine whether a packet will be seen by a targeted host and how it may be processed. The second is at the application layer where ManHunt's knowledge of application protocols and allowed encodings will result in the proper recognition of even obfuscated data.

Vulnerability Patterns

Symantec Security Response has had great success creating signatures for Symantec ManHunt 3.0 that detect attempts to exploit a vulnerability independent of the exploit tool used. Traditionally signatures have targeted specific tools. These new *vulnerability patterns* are able to detect new tools as soon as they hit the network.

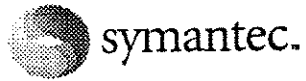
Recognizing this success and leveraging the expertise developed by four years of providing PAD, Symantec ManHunt 4.0 will add a new signature matching engine. This engine uses a highly expressive language to describe the extremely complex patterns associated with these vulnerability signatures.

This expressiveness will allow Symantec Security Response to release vulnerability signatures faster than ever. This results in earlier prevention of threats and more complete coverage.

Deception Insertion

Leveraging the expertise that developed Symantec Decoy Server, future versions of Symantec ManHunt will include basic deception capabilities as an additional detection device. This will allow the diversion and distraction of live attackers; the delay and detection of automated attackers; and a new depth of valuable forensic data, all with zero false positives.

This light-weight deception for network intrusion detection will provide early detection of known and unknown threats and insight into interactive and automated attackers. Like the current ability to craft TCP Reset packets to force an attacker to terminate sessions, the Symantec ManHunt deception module will create fictitious network devices, clients,



servers, and routers to disguise the real ones. Attackers will then attempt to probe and attack these decoys. This gives the ManHunt operator the time and information needed to respond before an attack has hit a real host.

Integration with Symantec ManHunt's custom signature capability will allow the rapid development and deployment of new signatures in response to new threats. By capturing the traffic targeted at decoy hosts and correlating that with the likely service behavior, ManHunt will provide the ideal template for a new signature.

Conclusion

Combining multiple methods of threat detection provides both broad and deep detection of network threats. Today Symantec ManHunt uses PAD, signatures and traffic analysis to detect a complete set of threats. Tomorrow new detection techniques will continue to provide state-of-the-art prevention capabilities.

CONFIDENTIAL